

CLAIMS

1 1. In a cluster of computing nodes having shared access
2 to one or more file systems in data storage using
3 parallel file system software, a method for managing the
4 data storage, comprising:

5 initiating a session of a data management
6 application on a first one of the nodes;

7 receiving a request submitted to the parallel file
8 system software at a second one of the nodes to mount one
9 of the file systems in the data storage on the second one
10 of the nodes; and

11 sending a mount event message from the second node
12 to the first node responsive to the request, for
13 processing by the data management application on the
14 first node.

1 2. A method according to claim 1, and comprising
2 mounting first and second instances of the one of the
3 file systems on the first and second nodes, respectively,
4 responsive to the mount event message.

1 3. A method according to claim 2, and comprising:

2 receiving a further request at the second node to
3 unmount the second instance of the one of the file
4 systems at the second node;

5 sending, responsive to the further request, a
6 preunmount event message to the first node; and

7 responding to the preunmount event message so as to
8 permit unmounting of the second file system instance
9 without unmounting the first file system instance.

1 4. A method according to claim 3, wherein responding to
2 the preunmount event message comprises determining at the

3 first node, responsive to one or more flags set in the
4 preunmount event message, whether the request was
5 submitted on the first node or on another one of the
6 nodes.

1 5. A method according to claim 3, and comprising:
2 receiving the preunmount event message at the first
3 node;

4 obtaining a data management access right from a
5 physical file system (PFS) software component at the
6 first node responsive to the preunmount event message;
7 and

8 processing the preunmount event message using the
9 access right.

1 6. A method according to claim 3, wherein receiving the
2 request comprises receiving first and second requests to
3 mount different ones of the file systems in the data
4 storage, and wherein receiving the further request
5 comprises receiving further first and second requests to
6 unmount the different ones of the file systems, and
7 wherein sending the preunmount event message comprises,
8 responsive to dispositions set for the different ones of
9 the file systems, sending a first preunmount event
10 message to the first node responsive to the first unmount
11 request, and sending a second preunmount event message
12 responsive to the second unmount request to a further
13 node, on which a further data management application
14 session has been initiated.

1 7. A method according to claim 3, wherein responding to
2 the preunmount event message comprises sending a reply to
3 the message from the first node to the second node, and
4 comprising, responsive to the reply, unmounting the

5 second file system instance and sending an unmount event
6 message from the second node to the first node.

1 8. A method according to claim 7, and comprising
2 determining at the first node, responsive to one or more
3 flags set in the unmount event message, whether the
4 further request was submitted on the first node or on
5 another one of the nodes.

1 9. A method according to claim 1, and comprising
2 determining at the first node, responsive to one or more
3 flags set in the mount event message, whether the request
4 was submitted on the first node or on another one of the
5 nodes.

1 10. A method according to claim 1, wherein initiating
2 the session comprises initiating the session in
3 accordance with a data management application programming
4 interface (DMAPI) of the parallel file system software,
5 and wherein receiving the request and sending the mount
6 event message comprise processing the request and sending
7 the message using the DMAPI.

1 11. A method according to claim 10, and comprising
2 receiving an unmount request to unmount the file system
3 from the second node using the DMAPI, and sending a
4 preunmount event message to the first node responsive to
5 the unmount request using the DMAPI, for processing by
6 the data management application on the first node.

1 12. A method according to claim 11, and comprising
2 sending a reply to the preunmount event message from the
3 first node to the second node using the DMAPI, and,
4 responsive to the reply, unmounting the file system at

5 the second node, and sending an unmount event message to
6 the first node using the DMAPI.

1 13. A method according to claim 10, and comprising
2 receiving and processing the event message at the first
3 node using one or more functions of the DMAPi called by
4 the data management application.

1 14. A method according to claim 10, wherein sending the
2 event message comprises sending the message for
3 processing in accordance with a disposition specified by
4 the data management application using the DMAPi for
5 association with an event generated by the file
6 operation.

1 15. A method according to claim 10, wherein sending the
2 event message comprises setting one or more flags in the
3 message to indicate whether the request was submitted on
4 the first node or on another one of the nodes.

1 16. A method according to claim 10, and comprising
2 invoking a function of the DMAPi to obtain mount
3 information regarding the one of the file systems, and
4 wherein in a response provided by the function, one or
5 more flags are set to indicate whether the one of the
6 file systems is mounted on the first node or on another
7 one of the nodes in the cluster or on both the first node
8 and on another one of the nodes in the cluster.

1 17. A method according to claim 1, and comprising:
2 receiving a response to the mount event message from
3 the data management application on the first node; and
4 mounting an instance of the one of the file systems
5 on the second node subject to the response from the data
6 management application on the first node.

1 18. A method according to claim 1, and comprising
2 receiving a further request submitted to the parallel
3 file system software to mount the one of the file systems
4 on a further one of the nodes, and sending a further
5 mount event message responsive to the further request for
6 processing by the data management application on the
7 first node.

1 19. A method according to claim 18, wherein the further
2 one of the nodes is the first node.

1 20. A method according to claim 19, and comprising
2 receiving first and second unmount requests to unmount
3 the file system from the second node and from the further
4 one of the nodes, and generating first and second
5 preunmount event messages at the second node and at the
6 further one of the nodes responsive to the first and
7 second unmount requests, for processing by the data
8 management application on the first node.

1 21. A method according to claim 20, and comprising
2 sending a reply to the first and second preunmount event
3 messages from the first node to the second node and to
4 the further one of the nodes, and, responsive to the
5 reply, unmounting the file system at the second node and
6 the further one of the nodes, and generating respective
7 unmount event messages at the second node and at the
8 further one of the nodes.

1 22. A method according to claim 1, wherein initiating
2 the session of the data management application comprises
3 initiating a data migration application, so as to free
4 storage space on at least one of the volumes of data
5 storage.

1 23. Computing apparatus, comprising: .
2 one or more volumes of data storage, arranged to
3 store data in one or more file systems; and
4 a plurality of computing nodes, linked to access the
5 volumes of data storage using parallel file system
6 software, and arranged so as to enable a data management
7 application to initiate a data management session on a
8 first one of the nodes, so that when a request is
9 submitted to the parallel file system software at a
10 second one of the nodes to mount one of the file systems
11 in the data storage on the second one of the nodes, a
12 mount event message is sent from the second node to the
13 first node responsive to the request, for processing by
14 the data management application on the first node.

1 24. Apparatus according to claim 23, wherein the nodes
2 are arranged so that first and second instances of the
3 one of the file systems are mounted on the first and
4 second nodes, respectively, responsive to the mount event
5 message.

1 25. Apparatus according to claim 24, wherein responsive
2 to a further request at the second node to unmount the
3 second instance of the one of the file systems at the
4 second node, a preunmount event message is sent to the
5 first node, which is arranged to respond to the
6 preunmount event message so as to permit unmounting of
7 the second file system instance without unmounting the
8 first file system instance.

1 26. Apparatus according to claim 25, wherein the first
2 node is arranged to respond to the unmount event message
3 by determining, responsive to one or more flags set in

4 the preunmount event message, whether the request was
5 submitted on the first node or on another one of the
6 nodes.

1 27. Apparatus according to claim 25, wherein the first
2 node is arranged, upon receiving the preunmount event
3 message, to obtain a data management access right from a
4 physical file system (PFS) software component at the
5 first node responsive to the preunmount event message,
6 and to process the preunmount event message using the
7 access right.

1 28. Apparatus according to claim 25, wherein the request
2 comprises first and second requests to mount different
3 ones of the file systems in the data storage, and wherein
4 the further request comprises further first and second
5 requests to unmount the different ones of the file
6 systems, and wherein the nodes are arranged, responsive
7 to dispositions set for the different ones of the file
8 systems, to send a first preunmount event message to the
9 first node responsive to the first unmount request, and
10 to send a second preunmount event message responsive to
11 the second unmount request to a further node, on which a
12 further data management application session has been
13 initiated.

1 29. Apparatus according to claim 25, wherein the first
2 node is arranged to send a reply to the message to the
3 second node, and responsive to the reply, the second node
4 is arranged to unmount the second file system instance
5 and to send an unmount event message to the first node.

1 30. Apparatus according to claim 29, wherein the first
2 node is arranged to determine, responsive to one or more

3 flags set in the unmount event message, whether the
4 further request was submitted on the first node or on
5 another one of the nodes.

1 31. Apparatus according to claim 23, wherein the first
2 node is arranged to determine, responsive to one or more
3 flags set in the mount event message, whether the request
4 was submitted on the first node or on another one of the
5 nodes.

1 32. Apparatus according to claim 23, wherein the session
2 is initiated in accordance with a data management
3 application programming interface (DMAPI) of the parallel
4 file system software, and wherein the request is
5 processed and the mount event message is sent using the
6 DMAPI.

1 33. Apparatus according to claim 32, wherein when an
2 unmount request is received to unmount the file system
3 from the second node using the DMAPI, a preunmount event
4 message is sent to the first node responsive to the
5 unmount request using the DMAPI, for processing by the
6 data management application on the first node.

1 34. Apparatus according to claim 33, wherein the first
2 node is arranged to send a reply to the preunmount event
3 message to the second node using the DMAPI, wherein
4 responsive to the reply, the file system is unmounted at
5 the second node, and an unmount event message is sent to
6 the first node using the DMAPI.

1 35. Apparatus according to claim 32, wherein the event
2 message is received and processed at the first node using
3 one or more functions of the DMAPI called by the data
4 management application.

1 36. Apparatus according to claim 32, wherein the mount
2 event message is sent for processing in accordance with a
3 disposition specified by the data management application
4 using the DMAPI for association with the mount event.

1 37. Apparatus according to claim 32, wherein one or more
2 flags are set in the event message to indicate whether
3 the request was submitted on the first node or on another
4 one of the nodes.

1 38. Apparatus according to claim 32, wherein the first
2 node is arranged to invoke a function of the DMAP to
3 obtain mount information regarding the one of the file
4 systems, and wherein in a response provided by the
5 function, one or more flags are set to indicate whether
6 the one of the file systems is mounted on the first node
7 or on another one of the nodes in the cluster or on both
8 the first node and on another one of the nodes in the
9 cluster.

1 39. Apparatus according to claim 23, wherein after the
2 mount event message is received at the first node, an
3 instance of the one of the file systems is mounted on the
4 second node subject to the response from the data
5 management application on the first node.

1 40. Apparatus according to claim 23, wherein responsive
2 to a further request submitted to the parallel file
3 system software to mount the one of the file systems on a
4 further one of the nodes, a further mount event message
5 responsive to the further request is sent for processing
6 by the data management application on the first node.

1 41. Apparatus according to claim 40, wherein the further
2 one of the nodes is the first node.

1 42. Apparatus according to claim 41, wherein upon
2 receiving first and second unmount requests to unmount
3 the file system from the second node and from the further
4 one of the nodes, first and second preunmount event
5 messages are generated at the second node and at the
6 further one of the nodes responsive to the first and
7 second unmount requests, for processing by the data
8 management application on the first node.

1 43. Apparatus according to claim 42, wherein the first
2 node is arranged to send a reply to the first and second
3 preunmount event messages to the second node and to the
4 further one of the nodes, and wherein, responsive to the
5 reply, the file system is unmounted at the second node
6 and the further one of the nodes, and respective unmount
7 event messages are generated at the second node and at
8 the further one of the nodes.

1 44. Apparatus according to claim 23, wherein the data
2 management application comprises a data migration
3 application, for freeing storage space on at least one of
4 the volumes of data storage.

1 45. A computer software product for use in a cluster of
2 computing nodes having shared access to one or file
3 systems in data storage, accessed using parallel file
4 system software, the product comprising a
5 computer-readable medium in which program instructions
6 are stored, which instructions, when read by the
7 computing nodes, cause a session of a data management
8 application to be initiated on a first one of the nodes,
9 and in response to a request submitted to the parallel
10 file system software at a second one of the nodes to

11 mount one of the file systems in the data storage on the
12 second node, cause the second node to send a mount event
13 message to the first node, for processing by the data
14 management application on the first node.

1 46. A product according to claim 45, wherein the
2 instructions cause the nodes to mount first and second
3 instances of the one of the file systems on the first and
4 second nodes, respectively, responsive to the mount event
5 message.

1 47. A product according to claim 46, wherein responsive
2 to a further request at the second node to unmount the
3 second instance of the one of the file systems at the
4 second node, the instructions cause a preunmount event
5 message to be sent to the first node, and cause the first
6 node to respond to the preunmount event message so as to
7 permit unmounting of the second file system instance
8 without unmounting the first file system instance.

1 48. A product according to claim 47, wherein the
2 instructions cause the first node to respond to the
3 preunmount event message by determining, responsive to
4 one or more flags set in the preunmount event message,
5 whether the request was submitted on the first node or on
6 another one of the nodes.

1 49. A product according to claim 47, wherein the
2 instructions cause the first node, upon receiving the
3 preunmount event message, to obtain a data management
4 access right from a physical file system (PFS) software
5 component at the first node responsive to the preunmount
6 event message, and to process the preunmount event
7 message using the access right.

1 50. A product according to claim 47, wherein the request
2 comprises first and second requests to mount different
3 ones of the file systems in the data storage, and wherein
4 the further request comprises further first and second
5 requests to umount the different ones of the file
6 systems, and wherein the instructions cause the nodes,
7 responsive to dispositions set for the different ones of
8 the file systems, to send a first preunmount event
9 message to the first node responsive to the first umount
10 request, and to send a second preunmount event message
11 responsive to the second umount request to a further
12 node, on which a further data management application
13 session has been initiated.

1 51. A product according to claim 47, wherein the
2 instructions cause the first node to send a reply to the
3 message to the second node, and cause the second node,
4 responsive to the reply, to umount the second file
5 system instance and to send an umount event message to
6 the first node.

1 52. A product according to claim 51, wherein the
2 instructions cause the first node to determine,
3 responsive to one or more flags set in the umount event
4 message, whether the further request was submitted on the
5 first node or on another one of the nodes.

1 53. A product according to claim 45, wherein the
2 instructions cause the first node to determine,
3 responsive to one or more flags set in the mount event
4 message, whether the request was submitted on the first
5 node or on another one of the nodes.

1 54. A product according to claim 45, wherein the product
2 comprises a data management application programming
3 interface (DMAPI) of the parallel file system software,
4 and wherein the request is processed and the mount event
5 message is sent using the DMAPI.

1 55. A product according to claim 54, wherein when an
2 unmount request is received to unmount the file system
3 from the second node using the DMAPI, the instructions
4 cause a preunmount event message to be sent to the first
5 node responsive to the unmount request using the DMAPI,
6 for processing by the data management application on the
7 first node.

1 56. A product according to claim 55, wherein the
2 instructions cause the first node to send a reply to the
3 preunmount event message to the second node using the
4 DMAPI, wherein responsive to the reply, the file system
5 is unmounted at the second node, and an unmount event
6 message is sent to the first node using the DMAPI.

1 57. A product according to claim 54, wherein the event
2 message is received and processed at the first node using
3 one or more functions of the DMAPI called by the data
4 management application.

1 58. A product according to claim 54, wherein the event
2 message is sent for processing in accordance with a
3 disposition specified by the data management application
4 using the DMAPI for association with an event generated
5 by the file system.

1 59. A product according to claim 54, wherein one or more
2 flags are set in the event message to indicate whether

3 the request was submitted on the first node or on another
4 one of the nodes.

1 60. A product according to claim 54, wherein the
2 instructions cause the first node to invoke a function of
3 the DMAPI to obtain mount information regarding the one
4 of the file systems, and wherein in a response provided
5 by the function, one or more flags are set to indicate
6 whether the one of the file systems is mounted on the
7 first node or on another one of the nodes in the cluster
8 or on both the first node and on another one of the nodes
9 in the cluster.

1 61. A product according to claim 45, wherein after the
2 mount event message is received at the first node, an
3 instance of the one of the file systems is mounted on the
4 second node subject to the response from the data
5 management application on the first node.

1 62. A product according to claim 45, wherein responsive
2 to a further request submitted to the parallel file
3 system software to mount the one of the file systems on a
4 further one of the nodes, a further mount event message
5 responsive to the further request is sent for processing
6 by the data management application on the first node.

1 63. A product according to claim 62, wherein the further
2 one of the nodes is the first node.

1 64. A product according to claim 63, wherein upon
2 receiving first and second unmount requests to unmount
3 the file system from the second node and from the further
4 one of the nodes, the instructions cause first and second
5 preunmount event messages to be generated at the second
6 node and at the further one of the nodes responsive to

7 the first and second unmount requests, for processing by
8 the data management application on the first node.

1 65. A product according to claim 64, wherein the
2 instructions cause the first node to send a reply to the
3 first and second preunmount event messages to the second
4 node and to the further one of the nodes, and wherein,
5 responsive to the reply, the file system is unmounted at
6 the second node and the further one of the nodes, and
7 respective unmount event messages are generated at the
8 second node and at the further one of the nodes.

1 66. A product according to claim 45, wherein the data
2 management application comprises a data migration
3 application, for freeing storage space on at least one of
4 the volumes of data storage.